to all memory components 190, and they drive the result back to the nested MBIST engine 620 on another direction test pattern bus. The results are stored in the MBIST signature register 605 as part of the MBIST signature 140. When the MBIST is completed, the MBIST state machine 610 transitions to the done state 850, signaling completion by setting the dedicated bit in the MBIST signature register 605 to indicate the MBIST is complete.

As was mentioned above, the nested MBIST engine 620 is, in the illustrated embodiment, a vendor-supplied MBIST engine such as vendors use in their testers. The states 810, 820, 830, 840, and 850 of the individual MBIST state machines 610 may be implemented in accordance with conventional practice. Furthermore, the operation of the MBIST state machines 610 will be implementation specific depending on the implementation of the nested MBIST engine 620.

More particularly, in the illustrated embodiment, the memory components 190 are SRAMs and the testing interface 180 is a JTAG tap ("JTTAP") implemented as is known in the art. The MBIST engine 120 is reset by asserting the external reset signal received through the testing interface 180. With the JTAG Tap (not shown) controller signals of MBST_SEL and MBST_RUN, the MBIST engine 120 is initialized. Initialization is followed by flush and then the test patterns as the MBIST engine 120 cycles through the initialization state 820, flush state 830, and test state 840. The flush state 830 occurs, in the illustrated embodiment, for 1024, 75 MHz cycles and initializes the SRAM to a known state. Flush state MUX gates (not shown) are hand-instantiated within the SRAM wrappers 910 to hold the SCAN_IN IO (on which the dual mode BIST controller 100 outputs scan patterns) to a 1'b0, the first and second scan clock signals are both held to a 1'b1 as the SRAM is flushed to all zeros. Watchdog timers (not shown) are part of paranoid logic in the MBIST engine 120 to prevent the nested MBIST engine 620 from free running or having any destructive effects during normal functionality. The MBIST engine 120 drives a one direction test pattern bus (not shown) out to all SRAMs, and the SRAMs drive the result back to the nested MBIST engine 620.

In operation, the ASIC 100 shown in **FIG. 1** may be placed on a vendor-supplied tester having a test controller 915 including a JTAG controller 920, shown in **FIG. 9**, typically with several other ASICs 100 (not shown). Alternatively, the ASIC 100 may be

tested in a live system having a live system controller 925 including a JTAG controller 920. The MBIST engine 120 includes a MBIST state machine 610, shown in **FIG. 6**, designed for use with this particular vendor-supplied test controller 915. In the illustrated embodiment, the JTAG controller 920 employs JTAG protocols and testing hardware, and so the testing interface 180 is a JTTAP controller. As was noted above, the LBIST and MBIST capabilities of the dual mode BIST controller 100 may be utilized separately or conjunctively. Furthermore, the LBIST and the MBIST may be performed in parallel or in serial. However, the following discussion will contemplate a conjunctive use in serial. It is nevertheless to be understood that only one or the other may be employed in alternative embodiments.

The JTAG controller 920, shown in **FIG. 9**, of the vendor-supplied test controller 915 or the live system controller 925 provides the configuration data including the vector count and the PRPG seed to the LBIST domain 160 through the testing interface 180. The testing interface 180, under the control of the JTAG controller 920, then supplies the external reset signal, shown in **FIG. 2** and **FIG. 6**, to the LBIST domain 160 and the MBIST domain 170. The LBIST state machine 210 and the MBIST state machine 610 then each transition to their respective reset states 310, 810.

The testing interface 180, again under control of the JTAG controller 920, generates the LBIST run signal, whereupon the LBIST state machine 320 transitions into the initiate state 320. The LBIST engine 110 then initiates as was discussed above. The LBIST state machine 110 then cycles through the scan and step states 330, 340 as discussed above until the LBIST is complete, *i.e.*, the value of the pattern generator 230 is equal to the configured vector count. As the LBIST is run, the results are stored in the MISR 220. Note that the LBIST is run at the slowest frequency in the testing interface 180 and the logic core 900, such that the results stored in the MISR 220 are free from errors that would otherwise arise from timing violations. When the LBIST is complete, the LBIST state machine 210 transitions to the done state 350. The LBIST engine 110 then generates a "complete" signal that sets a bit in the MISR 220 to indicate that the LBIST has successfully completed. If, for some reason, the pattern generator 230 goes to all zeroes, the error signal is instead generated and the LBIST aborted.

The testing interface 180 then generates the MBIST run and MBIST select signals, whereupon the MBIST state machine 610 transitions to the initialize state 820. The MBIST engine 120 initializes its components and signals as was discussed above. The MBIST state machine 610 then cycles through the flush and test states 830, 840 as discussed above using the nested MBIST engine 620. As the MBIST is run, the results of the paranoid checks and the MBIST engine states are stored in the MBIST signature register 605. When the MBIST is complete, the MBIST state machine 610 transitions to the done state 850, whereupon the MBIST engine 120 generates the complete signal, which sets a done bit in the MBIST signature register 605.

The dual mode BIST controller 100 permits all this functionality to be designed into a single module of the ASIC 150. This further facilitates the placement of other ASIC components and the wiring between them. The dual mode BIST controller 100 also permits the use of multiple clock domains in the same module. Because the results of both the LBIST and the MBIST are stored, the system controller 925 in the live system or the vendor-supplied test controller 915 can read out the results of the tests through the testing interface 180.

This concludes the detailed description. The particular embodiments disclosed above are illustrative only, as the invention may be modified and practiced in different but equivalent manners apparent to those skilled in the art having the benefit of the teachings herein. Furthermore, no limitations are intended to the details of construction or design herein shown, other than as described in the claims below. It is therefore evident that the particular embodiments disclosed above may be altered or modified and all such variations are considered within the scope and spirit of the invention. Accordingly, the protection sought herein is as set forth in the claims below.